

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A computer system comprising[[.]]:
a central processing unit (CPU); and
a first and second network adapter teamed together and configured to
receive offloaded connections; ~~and~~
wherein a program executing on the CPU reloads an offloaded connection
established by the first network adapter onto the second network
adapter [[if]] as a result of one of a plurality of packets associated
with the offloaded connection ~~was-being~~ received on the second
network adapter.
2. (Original) The system of claim 1 wherein the first and second network
adapters are capable of fully offloading all protocol processing.
3. (Original) The system of claim 1 wherein the first and second network
adapters transmit and receive packets of data using a single media access
control (MAC) and internet protocol (IP) address.
4. (Original) The system of claim 1 wherein the program reloads an
offloaded connection by transferring the context of the connection from the first
network adapter to the second network adapter.
5. (Original) The system of claim 1 wherein the program monitors every
packet received by the first and second network adapters and inactivates
connections associated with packets that have not been received for a defined
time period.

6. (Original) The system of claim 1 wherein the first and second network adapters send a notification to the program if a connection is prematurely terminated.

7. (Original) The system of claim 1 wherein the first and second network adapters comprise network interface cards (NICs).

8. (Currently amended) A method comprising:
examining a packet received from an external device;
determining whether a connection associated with the packet is currently
offloaded; and
reloading the connection [[if]] in response to the packet associated with
the connection [[is]] being offloaded and received by a network
interface not currently processing the offloaded connection.

9. (Original) The method of claim 8 further comprising determining an identifier for the network interface that receives the packet and writing the determined identifier to a memory.

10. (Original) The method of claim 8 wherein the reloading further comprises copying the context of the connection to the network interface that received the packet.

11. (Original) The method of claim 8 wherein the network interface that received the packet and the network interface currently offloading the connection are teamed together.

12. (Currently amended) A computer readable media storing instructions executable by a computer system, and when executed the instructions implement a method comprising:
examining a packet received from an external device;

determining whether a connection associated with the packet is currently offloaded; and
reloading the connection **[[if]]** as a result of the packet associated with the connection **[[is]]** being offloaded and received by a network interface not currently processing the offloaded connection.

13. (Original) The computer readable media of claim 12 further comprising determining an identifier for the network interface that receives the packet and writing the determined identifier to a memory unit.

14. (Original) The computer readable media of claim 12 wherein the reloading further comprises copying the context of the connection to the network interface that received the packet.

15. (Original) The computer readable media of claim 12 wherein the network interface that received the packet and the network interface currently offloading the connection are teamed together.

16. (Currently amended) A computer system comprising:

[[a]] means for reading and executing programs; and

[[a]] first and second means for sending and receiving data connections over a network, the **[[a]]** first and second means grouped together and capable of processing offloaded data connections;

wherein a program executed by the means for reading and executing programs reloads an offloaded connection established by the first means for sending and receiving data onto the second means for sending and receiving data **[[if]]** in response to one of a plurality of packets associated with the offloaded connection ~~was~~ being received on the second means for sending and receiving data.

17. (Original) The system of claim 16 wherein the first and second means for sending and receiving data connections are capable of fully offloading all protocol processing.

18. (Original) The system of claim 16 wherein the first and second means for sending and receiving data connections send and receive packets of data using a single media access control (MAC) and internet protocol (IP) address.

19. (Original) The system of claim 16 wherein the program reloads an offloaded connection by transferring the context of the connection from the first means for sending and receiving data connections to the second means for sending and receiving data connections.

20. (Original) The system of claim 16 wherein the program monitors all data received by the first and second means for sending and receiving data connections.

21. (Original) The system of claim 16 wherein the first and second means for sending and receiving data connections send a notification to the program if a connection is prematurely terminated.